An Upper Bound on the Cutoff Rate of Sequential Decoding

ERDAL ARIKAN, MEMBER, IEEE

Abstract—**An upper bound is given on the cutoff rate of discrete memoryless channels. This upper bound, which coincides with a known lower bound, determines the cutoff rate, and settles a long-standing open problem.**

I. Introduction

A. The Problem

Sequential decoding is a decoding algorithm for tree codes invented by Wozencraft [1] and later developed by Fano [2]. In essence, sequential decoding is a search algorithm for finding that path in a tree code which corresponds to the encoded message. The complexity of this algorithm, which can be defined roughly as the number of computations per correctly decoded source digit, is a random variable. For obvious reasons, sequential decoding is considered impractical in a given situation if its average complexity is unbounded. Without this constraint on the average decoding complexity, sequential decoding can be used at rates up to channel capacity, yielding a probability of decoding error as low as desired. With this constraint, however, the maximum achievable rate is typically strictly lower than the capacity. This maximum rate is called the (computational) cutoff rate of sequential decoding, and is denoted by R_{comp}.

It is well-known [3, p. 279] that $R_{\text{comp}}(K) \geq R_0(K)$ for every discrete memoryless channel (DMC) K, where

$$R_0(K) = \max_Q -\ln \sum_{\eta \in Y} \left\{ \sum_{\xi \in X} Q(\xi)\sqrt{P(\eta|\xi)} \right\}^2. \quad (1.1)$$

Here, X denotes the input alphabet, Y the output alphabet, and P the transition probabilities of K; the maximum is taken over all probability distributions (p.d.'s) on X. In the following, we shall write $K = (P, X, Y)$ to denote a channel with these parameters. It will be assumed throughout that the channel input and output alphabets are finite.

Manuscript received December 15, 1985; revised November 6, 1986. The research for this work was conducted at M.I.T. Laboratory for Information and Decision Systems and supported by Defence Advanced Research Projects Agency under Contract N000 14-84-K-0357.

The author is with the Department of Electrical Engineering, Bilkent University, P.K. 8, Maltepe, Ankara, 06572, Turkey.

IEEE Log Number 8718720.

This paper proves that $R_{\text{comp}}(K) \leq R_0(K)$ for every DMC K, and thus determines the cutoff rate of DMC's.

B. Previous Work

Jacobs and Berlekamp [4] studied the complexity of sequential decoding and made several significant contributions. To state one of their results which is relevant here, we need some definitions. Let C_n be $(1/n)$ times the number of computations performed by the sequential decoding algorithm to correctly decode the first n symbols of the message sequence. For any DMC $K = (P, X, Y)$ and $\rho > 0$, let[1]

$$E_0(K, \rho) = \max_Q -\ln \sum_{\eta \in Y} \left\{ \sum_{\xi \in X} Q(\xi) P(\eta|\xi)^{1/(1+\rho)} \right\}^{(1+\rho)} \quad (1.2)$$

where the maximum is computed over all p.d.'s on X. Finally, let $\hat{E}_0(K, \rho)$ be the smallest concave function greater than or equal to $E_0(K, \rho)$. Jacobs and Berlekamp proved that if the code rate exceeds $\hat{E}_0(K, \rho)/\rho$, then

$$\lim_{n \to \infty} E(C_n^\rho) = \infty. \quad (1.3)$$

In particular, by setting $\rho = 1$, the above result implies that $R_{\text{comp}}(K) \leq \hat{E}_0(K, 1)$. Since $E_0(K, 1) = R_0(K)$ and $R_0(K) \leq R_{\text{comp}}(K)$, it follows that $R_{\text{comp}}(K) = R_0(K)$ for all K for which $\hat{E}_0(K, 1) = E_0(K, 1)$. Here, we extend this result to channels for which $E_0(K, 1) < \hat{E}_0(K, 1)$. Such channels do exist, as the next example illustrates.

Example:[2] Let the transition probabilities be as in Fig. 1. The function $E_0(K, \rho)$ is also given in Fig. 1. Observe that there is a slope discontinuity at (around) $\rho = 1$, and $E_0(K, 1) < \hat{E}_0(K, 1)$.

Savage [5] also studied the problem of computation in sequential decoding, and conjectured that (1.3) holds whenever the rate exceeds $E_0(K, \rho)/\rho$. Our result establishes Savage's conjecture for $\rho = 1$, but in its full generality the conjecture still remains open.

[1] The function E_0 is known as Gallager's reliability exponent [3, p. 143].

[2] This example is essentially identical to [3, Example 2, p. 147], where it is also shown how the function E_0 can be computed.

0018-9448/88/0100-0055$01.00

	output			
	1	2	3	4
1	.400	.200	.200	.200
2	.200	.400	.200	.200
input 3	.200	.200	.400	.200
4	.200	.200	.200	.400
5	.329	.329	.171	.171
6	.171	.171	.329	.329

Transition probabilities

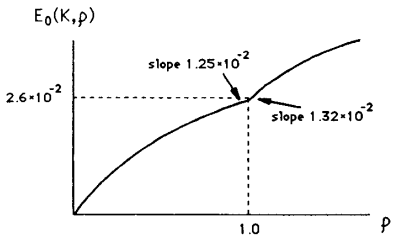

Fig. 1. Example of channel for which $E_0(K,1) < \hat{E}_0(K,1)$.

C. Outline of the Paper

This paper is organized as follows. Section II introduces the notation and terminology for tree codes, and briefly describes sequential decoding. Sections III and IV reduce the problem of lower-bounding the expected decoding complexity in sequential decoding to one of lower-bounding the probability of decoding error for block codes. Section V is a collection of known results about sphere-packing lower bounds to the probability of decoding error for fixed-composition block codes. In Section VI, we finally prove that $R_0(K) \geq R_{\text{comp}}(K)$ for every DMC K. In section VII, we interpret the results and point out a fundamental property of R_0.

The main results of this paper are Lemma 4.1 and Lemma 6.1.

II. Preliminaries

A. Basic Concepts and Notation

An encoder is a device which periodically receives a block of source digits and in response generates a block of channel input digits. We denote the mth source output (or encoder input) block by $\boldsymbol{u}(m)$, $m \geq 1$, and the source output sequence by $\boldsymbol{u} = \boldsymbol{u}(1), \boldsymbol{u}(2), \cdots$. We denote the mth encoder output (or channel input) block in response to a source sequence $\boldsymbol{u}$ by $\boldsymbol{eu}(m)$, and the encoder output sequence by $\boldsymbol{eu} = \boldsymbol{eu}(1), \boldsymbol{eu}(2), \cdots$. We denote the initial segments of these sequences by $\boldsymbol{u}(..m) = \boldsymbol{u}(1), \cdots, \boldsymbol{u}(m)$ and $\boldsymbol{eu}(..m) = \boldsymbol{eu}(1), \cdots, \boldsymbol{eu}(m)$.

The alphabet which $\boldsymbol{u}(m)$, $m \geq 1$, belongs to is called the source output (or encoder input) alphabet. For our purposes, there is no loss of generality in assuming that the source alphabet equals $\{0, \cdots, S-1\}$ for some integer S. The alphabet which $\boldsymbol{eu}(m)$, $m \geq 1$, belongs to is called the encoder output alphabet. If X is the channel input alphabet, then the encoder output alphabet equals X^k (the kth Cartesian power of X) for some integer k. An encoder with these parameters will be referred to as an (S, X, k) encoder in what follows.

An encoder $\boldsymbol{e}$ is called a *tree encoder* if $\boldsymbol{eu}(..m)$ depends only on $\boldsymbol{u}(..m)$ for each m. The mapping generated by a tree encoder is called a *tree code*. Further terminology will be introduced with the help of an example.

Example: Consider an encoder $\boldsymbol{e}$ with parameters $(2, \{0,1\}, 2)$ such that

$$\boldsymbol{eu}(m) = (\boldsymbol{u}(m-1) + \boldsymbol{u}(m), \boldsymbol{u}(m)), \qquad m \geq 1,$$

where + denotes addition modulo 2, and we arbitrarily set $\boldsymbol{u}(0) = 0$. The first three levels of the code tree for $\boldsymbol{e}$ are shown in Fig. 2. The tree representation is based on establishing a one-to-one mapping from source sequences to paths in the tree. In the present example, the mapping is indicated by the arrows at the left side of the diagram. In order to generate the encoded sequence, the encoder uses the source output as a sequence of instructions and follows the "upper" or the "lower" branch emanating to the right from the current node depending on whether the next source digit is, respectively, a 0 or a 1.

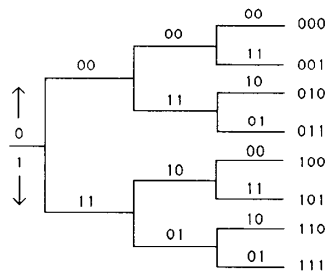

Fig. 2. Example of tree code.

For example, if the first three digits of the source output are 0,1,0, then the first three blocks (branches) of the encoded sequence are 00,11,10. Thus, each source sequence is mapped to a unique path. Hence, we refer to source sequences as *paths* and to initial segments of source sequences as *nodes*. For any path $\boldsymbol{u}$, and any $m = 1, 2, \cdots$, the *branch* connecting node $\boldsymbol{u}(..m-1)$ (for $m = 1$, take $\boldsymbol{u}(..m-1)$ as the origin) to node $\boldsymbol{u}(..m)$ is labeled by $\boldsymbol{eu}(m)$.

In the tree representation of an (S, X, k) tree code, each node at each level is connected to S nodes at the next higher level, and each branch is labeled by a block of k digits from X. The *rate* of such a code is defined as $(1/k)\ln S$ (nats/channel use) or $(1/k)\log_2 S$ (bits/channel use). All rates in this paper are in natural units (nats).

The path in the code tree that corresponds to the actual encoder output sequence (i.e., the transmitted sequence) is called the *correct path*. Nodes on the correct path are called *correct nodes*.

We adopt the following notation for the channel output sequence. The channel output block received in response to the mth channel input block $\boldsymbol{eu}(m)$ is denoted by $\boldsymbol{y}(m)$, the entire channel output sequence $\boldsymbol{y}(1), \boldsymbol{y}(2), \cdots$ by $\boldsymbol{y}$, and the initial segment $\boldsymbol{y}(1), \cdots, \boldsymbol{y}(m)$ by $\boldsymbol{y}(..m)$.

B. Sequential Decoding

Sequential decoding is a tree search algorithm for finding the correct path in a code tree based on information available from the channel output sequence. The algorithm relies on what is called a *metric* for directing its search.[3] Ordinarily, the metric is chosen as a function that measures the correlation between channel input and output sequences. However, any function Γ of the form

$$\Gamma: \bigcup_{m=1}^{\infty} X^{mk} \times Y^{mk} \to [-\infty, +\infty) \tag{2.1}$$

can serve as a metric in a situation where X is the channel input alphabet, Y the channel output alphabet, and k the number of channel input symbols per branch of the tree code. For example, the metric value of a node $\boldsymbol{u}(..m)$ is given by $\Gamma(\boldsymbol{eu}(..m), \boldsymbol{y}(..m))$.

Notice that $\Gamma(\boldsymbol{eu}(..m), \boldsymbol{y}(..m))$ does not depend on the portion of $\boldsymbol{y}$ beyond $\boldsymbol{y}(..m)$, namely $y(m+1)$, $y(m+2), \cdots$. This restriction on the form of metrics is an integral part of sequential decoding; without it, the upper bound of this paper on the cutoff rate would not hold. Also notice that the metric is allowed to take on the value $-\infty$. This makes it possible to rule out a node permanently from further consideration by the sequential decoding algorithm when there is no doubt that it is incorrect.

A well-known metric for sequential decoding is the following metric due to Fano [2], which is stated here as an example.

$$\Gamma(\boldsymbol{eu}(..m), \boldsymbol{y}(..m)) = \sum_{h=1}^{m} \left\{ \ln \frac{P(y(h)|\boldsymbol{eu}(h))}{\omega(y(h))} - \ln M \right\} \tag{2.2}$$

where ω is a p.d. on Y^k.

There are two well-known versions of sequential decoding: Fano's algorithm and the stack algorithm of Zigangirov [6] and Jelinek [7]. For practical purposes, Fano's algorithm is probably preferable since it requires almost no storage. However, in this paper we consider only the stack algorithm, primarily because it is simpler to describe and analyze. The results can easily be extended to Fano's algorithm.

At each step of the stack algorithm, there is a list of nodes in which nodes are ordered with respect to their metric values. This list is referred to as the *stack*. The metric values of the nodes in the stack increase towards the *top* of the stack. Ties between the metric values are broken by some fixed but arbitrary rule. Each step of the stack algorithm consists of deleting the node at the stack-top and inserting its immediate descendants into the stack. At the start of the algorithm, the origin is the only node in the stack, and it has a metric value of zero.

The tie-breaking rule mentioned above is assumed to be based on some ordering relation on the set of nodes in the code tree. Thus, for example, if $\boldsymbol{u}(..i)$ and $\boldsymbol{v}(..j)$ are any two nodes in the stack with equal metric values, $\boldsymbol{u}(..i)$ will be closer to the stack-top than $\boldsymbol{v}(..j)$ iff $\boldsymbol{u}(..i)$ precedes $\boldsymbol{v}(..j)$ with respect to this ordering relation.

To study the average complexity of sequential decoding, one has to make an assumption about the source statistics. Throughout, our assumption will be that each path in the code tree is equally likely to be the correct path. We shall express the average complexity of sequential decoding in terms of a quantity $\Lambda(K, \boldsymbol{e}, \Gamma, t)$ which we define as the expected number of nodes in a tree code $\boldsymbol{e}$ that reach the stack-top before the correct level-t node, assuming that K is the channel and Γ is the metric. Thus, $(1/t)\Lambda(K, \boldsymbol{e}, \Gamma, t)$ can be thought of as the average number of computations for the sequential decoder to move one step forward on the correct path along an initial segment consisting of t branches.

Intuitively speaking, sequential decoding cannot be considered practical if $(1/t)\Lambda(K, \boldsymbol{e}, \Gamma, t)$ goes to infinity as t increases. In fact, we would like to have $(1/t)\Lambda(K, \boldsymbol{e}, \Gamma, t)$ bounded uniformly over all t. Formalizing this idea, we say that a rate R is *achievable* (by sequential decoding) iff there exists a code $\boldsymbol{e}$ with rate $\geq R$ and a metric Γ such that

$$\sup_{t \geq 1} \{(1/t)\Lambda(K, \boldsymbol{e}, \Gamma, t)\} < \infty. \tag{2.3}$$

The supremum of all achievable rates is called the (computational) *cutoff rate* and denoted $R_{\text{comp}}(K)$.

The proof of $R_{\text{comp}}(K) \leq R_0(K)$ consists of showing that, for any rate R satisfying $R > R_0(K)$, it is impossible to find a code $\boldsymbol{e}$ with rate $\geq R$ and a metric Γ such that (2.3) holds. The first step of the proof will be to find a lower bound on Λ, which we do in the next section. We end this section by examining the necessary conditions for a node to reach the stack-top before some other node.

Consider sequential decoding of a code $\boldsymbol{e}$. Let Γ be the metric, $\boldsymbol{u}(..i)$ and $\boldsymbol{v}(..j)$ be any two distinct nodes, and $\boldsymbol{y}$ be the received sequence. Suppose that the minimum value of the metric on the path to node $\boldsymbol{u}(..i)$ is greater than that on the path to $\boldsymbol{v}(..j)$; i.e.,

$$\min_{1 \leq h \leq i} \{\Gamma(\boldsymbol{eu}(..h), \boldsymbol{y}(..h))\} > \min_{1 \leq h \leq j} \{\Gamma(\boldsymbol{ev}(..h), \boldsymbol{y}(..h))\}. \tag{2.4}$$

It follows directly from the rules of the stack algorithm that $\boldsymbol{v}(..j)$ cannot reach the stack-top before $\boldsymbol{u}(..i)$. There is, of course, no assertion that $\boldsymbol{u}(..i)$ will necessarily reach the stack-top. Notice that the tie-breaking rule plays no role in this argument.

Now suppose that

$$\min_{1 \leq h \leq i} \{\Gamma(\boldsymbol{eu}(..h), \boldsymbol{y}(..h))\} = \min_{1 \leq h \leq j} \{\Gamma(\boldsymbol{ev}(..h), \boldsymbol{y}(..h))\}. \tag{2.5}$$

In this case, it is the tie-breaking rule that determines which of the nodes $\boldsymbol{u}(..i)$ and $\boldsymbol{v}(..j)$ has priority over the other in reaching the stack-top. Again, there is no assertion that either node will necessarily reach the stack-top.

[3] The metric in sequential decoding is not a metric in the usual mathematical sense of the word.

We say that $\boldsymbol{u}(..i)$ is *favored over* $\boldsymbol{v}(..j)$ *when* $\boldsymbol{y}$ *is received* if $\boldsymbol{v}(..j)$ cannot reach the stack-top before $\boldsymbol{u}(..i)$ when $\boldsymbol{y}$ is the received sequence. Given two nodes $\boldsymbol{u}(..i)$ and $\boldsymbol{v}(..j)$, and a received sequence $\boldsymbol{y}$, one can determine which node is favored by running the stack algorithm for the pruned code tree in which all nodes are deleted except for those on the paths to $\boldsymbol{u}(..i)$ and $\boldsymbol{v}(..j)$. Notice that it suffices to know only an initial segment $\boldsymbol{y}(..\max\{i,j\})$ of the received sequence $\boldsymbol{y}$ to determine whether $\boldsymbol{u}(..i)$ is favored over $\boldsymbol{v}(..j)$ or not.

III. A Lower Bound on $\Lambda(K, \boldsymbol{e}, \Gamma, t)$

For and channel $K=(P, X, Y)$ and any block code $\boldsymbol{f}$ with block length N and codewords $\boldsymbol{f}(1), \cdots, \boldsymbol{f}(M)$, define

$$\lambda(K, \boldsymbol{f}) = (1/M) \sum_{i=1}^{M} \sum_{j=1}^{M} P(\mathscr{B}(i,j)|\boldsymbol{f}(i)) \quad (3.1)$$

where for each i and j,

$$\mathscr{B}(i,j) = \begin{cases} \{\eta \in Y^N\colon P(\eta|\boldsymbol{f}(i)) \le P(\eta|\boldsymbol{f}(j))\}, \\ \qquad\qquad \text{if } i \ne j, \\ \varnothing, \quad \text{if } i = j. \end{cases} \quad (3.2)$$

Note that $P(\mathscr{B}(i,j)|\boldsymbol{f}(i))$ is the probability that message j is at least as likely as message i, conditional on i being the transmitted message. If the messages are equiprobable, then $\lambda(K, \boldsymbol{f})$ is the expected number of incorrect messages that are at least as likely as the correct message.

The following lemma reduces the problem of lower-bounding $\Lambda(K, \boldsymbol{e}, \Gamma, t)$ to one of lower-bounding $\lambda(K, \boldsymbol{e}(t))$, where $\boldsymbol{e}(t)$ denotes the block code obtained by truncating the tree code $\boldsymbol{e}$ at level t. This idea has also been used by Jacobs and Berlekamp [4].

Lemma 3.1: $\Lambda(K, \boldsymbol{e}, \Gamma, t) \ge (1/2)\lambda(K, \boldsymbol{e}(t))$.

Proof: Let $K=(P, X, Y)$ and (S, X, k) be the parameters of $\boldsymbol{e}$. Let the level-t nodes in $\boldsymbol{e}$ be labeled by integers $1, \cdots, S^t$. Let $\boldsymbol{e}(t,i)$ denote the encoded sequence for the ith level-t node. $\boldsymbol{e}(t,i)$ will also denote the ith codeword of the block code $\boldsymbol{e}(t)$.

Claim:

$$\Lambda(K, \boldsymbol{e}, \Gamma, t) \ge (1/S^t) \sum_{i=1}^{S^t} \sum_{j=1}^{S^t} P(\mathscr{A}(i,j)|\boldsymbol{e}(t,i)) \quad (3.3)$$

where for each pair of distinct level-t nodes t and j, we have defined

$$\mathscr{A}(i,j) = \{\eta \in Y^{kt}\colon \text{node } j \text{ is favored over node } i \text{ when the first } t \text{ branches of the received sequence is } \eta\},$$

and $\mathscr{A}(i,i)=\varnothing$. Note that $\mathscr{A}(i,j)$ and $\mathscr{A}(j,i)$ are complementary sets in Y^{kt} for $i \ne j$.

Proof of the Claim: If the probability that the correct node at level t never reaches the stack-top is positive, then $\Lambda(K, \boldsymbol{e}, \Gamma, t)$ is infinite. So, without loss of generality, we may assume that $\boldsymbol{e}$ and Γ are such that the correct node at level t reaches the stack-top with probability one.

Suppose that node i is the correct node at level t. Let j be some other level-t node. Since i, being the correct node, reaches the stack-top with certainty, the probability that j reaches the stack-top before i equals $P(\mathscr{A}(i,j)|\boldsymbol{e}(t,i))$. Thus,

$$\sum_{j=1}^{S^t} P(\mathscr{A}(i,j)|\boldsymbol{e}(t,i)) \quad (3.4)$$

is the expected number of level-t nodes that reach the stack-top before node i, conditional on i being correct. Averaging (3.4) over i, we obtain (3.3).

The proof of Lemma 3.1 is now completed as follows. From (3.3), we have

$$2\Lambda(K, \boldsymbol{e}, \Gamma, t) \ge (1/S^t) \sum_{i=1}^{S^t} \sum_{j=1}^{S^t} \{P(\mathscr{A}(i,j)|\boldsymbol{e}(t,i)) + P(\mathscr{A}(j,i)|\boldsymbol{e}(t,j))\}. \quad (3.5)$$

Let us examine the summand in (3.5) for $i \ne j$.

$$\begin{aligned} &P(\mathscr{A}(i,j)|\boldsymbol{e}(t,i)) + P(\mathscr{A}(j,i)|\boldsymbol{e}(t,j)) \\ &\quad = \sum_{\eta \in \mathscr{A}(i,j)} P(\eta|\boldsymbol{e}(t,i)) + \sum_{\eta \in \mathscr{A}(j,i)} P(\eta|\boldsymbol{e}(t,j)) \\ &\quad \ge \sum_{\eta \in Y^{kt}} \min\{P(\eta|\boldsymbol{e}(t,i)), P(\eta|\boldsymbol{e}(t,j))\} \quad (3.5) \\ &\quad \ge (1/2)\{P(\mathscr{B}(i,j)|\boldsymbol{e}(t,i)) + P(\mathscr{B}(j,i)|\boldsymbol{e}(t,j))\}. \quad (3.6) \end{aligned}$$

Inequality (3.5) follows from the fact that $\mathscr{A}(i,j)$ and $\mathscr{A}(j,i)$ are complementary sets for $i \ne j$. The factor of $1/2$ in (3.6) accounts for the fact that, for $i \ne j$, $\mathscr{B}(i,j)$ and $\mathscr{B}(j,i)$ have in common those η for which $P(\eta|\boldsymbol{e}(t,i)) = P(\eta|\boldsymbol{e}(t,j))$.

Substitution of the above inequality into (3.5) yields the desired result, as shown below. Note that the $i=j$ terms in (3.7) are zero since $\mathscr{B}(i,i)=\varnothing$.

$$\begin{aligned} 2\Lambda(K, \boldsymbol{e}, \Gamma, t) &\ge (1/2S^t) \sum_{i=1}^{S^t} \sum_{j=1}^{S^t} \{P(\mathscr{B}(i,j)|\boldsymbol{e}(t,i)) + P(\mathscr{B}(j,i)|\boldsymbol{e}(t,j))\} \quad (3.7) \\ &= (1/S^t) \sum_{i=1}^{S^t} \sum_{j=1}^{S^t} P(\mathscr{B}(i,j)|\boldsymbol{e}(t,i)) \quad (3.8) \\ &= \lambda(K, \boldsymbol{e}(t)). \quad (3.9) \end{aligned}$$

IV. A Lower Bound on $\lambda(K, \boldsymbol{f})$

The purpose of this section is to lower-bound $\lambda(K, \boldsymbol{f})$ in terms of lower bounds to the probability of decoding error for fixed-composition block codes. We begin with some definitions.

A p.d. Q on X is said to be the *composition* of $\xi \in X^N$ iff, for each $\xi \in X$, $NQ(\xi)$ equals the number of times ξ appears in ξ. A p.d. Q on X is said to be a *composition class* on X^N iff $NQ(\xi)$ is integer-valued for each $\xi \in X$. A block code is said to be a *fixed-composition* block code iff all of its codewords have the same composition.

Let $K = (P, X, Y)$ be a DMC, and $\boldsymbol{f}$ be a block code for K with block length N and number of codewords M. Denote the codewords of $\boldsymbol{f}$ by $\boldsymbol{f}(1), \cdots, \boldsymbol{f}(M)$. In connection with K and $\boldsymbol{f}$, consider a decoder $\boldsymbol{d}$, and let z denote the output of $\boldsymbol{d}$. The probability of decoding error for message i is defined as $\Pr\{z \neq i | \boldsymbol{f}(i)\}$; i.e., the conditional probability that the decoder output is not equal to i given that $\boldsymbol{f}(i)$ is transmitted. The average probability of decoding error for a code $\boldsymbol{f}$ and decoder $\boldsymbol{d}$ is defined as

$$P_e(K, \boldsymbol{f}, \boldsymbol{d}) = (1/M) \sum_{i=1}^{M} \Pr\{z \neq i | \boldsymbol{f}(i)\}. \quad (4.1)$$

The above definitions are valid whether or not $\boldsymbol{f}$ is a fixed-composition code. For every composition class Q on X^N, we define

$$P_e(K, M, N, Q) = \min P_e(K, \boldsymbol{f}, \boldsymbol{d}) \quad (4.2)$$

where the minimum is over all codes $\boldsymbol{f}$ with M codewords, block length N, and fixed-composition Q, and all decoders $\boldsymbol{d}$.

The following lemma is the main result of this section; it is stated in a form slightly more general than is actually needed for our purposes.

Lemma 4.1: For every DMC $K = (P, X, Y)$, every code $\boldsymbol{f}$ with fixed-composition Q, block length N, and number of codewords M, and every collection of integers t, $M_1, \cdots, M_t$ satisfying 1) $t \geq 1$, (2) $M_i \geq 1$ for each $i = 1, \cdots, t$, and 3) $M - 1 = \sum_{1 \leq i \leq t}(M_i - 1)$,

$$\lambda(K, \boldsymbol{f}) \geq P_e(K, M_1, N, Q) + \cdots + P_e(K, M_t, N, Q). \quad (4.3)$$

Remark: To gain an intuitive feeling for this lemma, suppose that M/t is much larger than 1, and consider the case $M_i \approx M/t$ for each i. The above claim, which is now $\lambda(K, \boldsymbol{f}) \geq tP_e(K, M/t, N, Q)$, can be made plausible by considering the following experiment.

First randomly choose a message for transmission, then randomly partition the remaining messages into t groups such that the size of each group is $\approx M/t$. Transmit the codeword corresponding to the chosen message. Search each group for a message whose codeword is at least as likely, conditional on the received word, as the transmitted codeword; if there is such a message, say that an *error* has occurred in that group.

Lemma 4.1 lower-bounds $\lambda(K, \boldsymbol{f})$ by the expected number of groups in which errors occur. This expected number is simply the sum, over all groups, of the probability that there is an error in a given group. We expect to have the probability of error in each group to be $\geq P_e(K, M/t, N, Q)$, so we should have $\lambda(K, \boldsymbol{f}) \geq tP_e(K, M/t, N, Q)$, as the lemma claims.

The difficulty with this heuristic argument is that the groups in the above experiment do not stay fixed, whereas the quantity $P_e(K, M/t, N, Q)$ pertains to fixed codes. Nonetheless, the above ideas motivate the following proof. The proof is given for the general case (not just for $M_i \approx M/t$), but this generalization requires no new ideas.

Proof: Fix K, $\boldsymbol{f}$, and $M_1, \cdots, M_t$. Let $\boldsymbol{f}(1), \cdots, \boldsymbol{f}(M)$ be the codewords of $\boldsymbol{f}$. For each $i \in \{1, \cdots, M\}$, define

$$\mathscr{P}_i = \left\{ (S_1, \cdots, S_t): \bigcup_{j=1}^{t} S_j = \{1, \cdots, M\},\ i \in S_j,\ |S_j| = M_j,\ j = 1, \cdots, t \right\}.$$

Thus, if $(S_1, \cdots, S_t) \in \mathscr{P}_i$, then the sets $S_1, \cdots, S_t$ each contain i, but otherwise they are disjoint. (These sets correspond to the "groups" in the preceding heuristic argument.)

For $T \subset \{1, \cdots, M\}$, define $\mathscr{E}_i(T) = \{\eta \in Y^N$: There exists a $j \in T$, $j \neq i$, such that $P(\eta | \boldsymbol{f}(i)) \leq P(\eta | \boldsymbol{f}(j))\}$. ($\mathscr{E}_i(T)$ corresponds to the "error" event in "group" T given that the "chosen message" is i.)

Observe that, for any $\boldsymbol{S} = (S_1, \cdots, S_t) \in \mathscr{P}_i$,

$$\sum_{j=1}^{M} P(\mathscr{B}(i, j) | \boldsymbol{f}(i)) \geq \sum_{k=1}^{t} P(\mathscr{E}_i(S_k) | \boldsymbol{f}(i)) \quad (4.4)$$

where $\mathscr{B}(i, j)$ is as defined in (3.2). So, for any p.d. W_i on $\mathscr{P}_i$,

$$\sum_{j=1}^{M} P(\mathscr{B}(i, j) | \boldsymbol{f}(i)) \geq \sum_{\boldsymbol{S} \in \mathscr{P}_i} W_i(\boldsymbol{S}) \sum_{k=1}^{t} P(\mathscr{E}_i(S_k) | \boldsymbol{f}(i)). \quad (4.5)$$

Averaging both sides of (4.5) over i, we obtain

$$\lambda(K, \boldsymbol{f}) \geq (1/M) \sum_{i=1}^{M} \sum_{\boldsymbol{S} \in \mathscr{P}_i} W_i(\boldsymbol{S}) \sum_{k=1}^{t} P(\mathscr{E}_i(S_k) | \boldsymbol{f}(i)). \quad (4.6)$$

Now, let W_i in (4.6) be the uniform distribution, i.e., $W_i(\boldsymbol{S}) = 1/c$, where $c = (M-1)!/\{(M_1 - 1)! \cdots (M_t - 1)!\}$ is the cardinality of $\mathscr{P}_i$, and define

$$\alpha_k = (1/cM) \sum_{i=1}^{M} \sum_{\boldsymbol{S} \in \mathscr{P}_i} P(\mathscr{E}_i(S_k) | \boldsymbol{f}(i)) \quad (4.7)$$

to obtain

$$\lambda(K, \boldsymbol{f}) \geq \alpha_1 + \cdots + \alpha_t. \quad (4.8)$$

The following self-explanatory sequence of equations shows that $\alpha_k \geq P_e(K, M_k, N, Q)$ and completes the proof. The sets $\mathscr{F}(m)$ and $\mathscr{F}_i(m)$ that appear below are defined

as follows:

$$\mathscr{F}(m) = \{ D\colon D \subset \{1,\cdots,M\} \text{ and } D \text{ has } m \text{ elements}\}$$

$$\mathscr{F}_i(m) = \{ D \in \mathscr{F}(m)\colon i \in D\}.$$

$$\alpha_k = (1/cM) \sum_{i=1}^{M} \sum_{S \in \mathscr{P}_i} P(\mathscr{E}_i(S_k)|f(i)) \tag{4.9}$$

$$= (1/cM) \sum_{i=1}^{M} \sum_{D \in \mathscr{F}_i(M_k)} \sum_{S \in \mathscr{P}_i\colon S_k = D} P(\mathscr{E}_i(S_k)|f(i)) \tag{4.10}$$

$$= (1/cM) \sum_{i=1}^{M} \sum_{D \in \mathscr{F}_i(M_k)} P(\mathscr{E}_i(D)|f(i)) \cdot \sum_{S \in \mathscr{P}_i\colon S_k = D} 1 \tag{4.11}$$

$$= (1/cM) \sum_{i=1}^{M} \sum_{D \in \mathscr{F}_i(M_k)} P(\mathscr{E}_i(D)|f(i)) \cdot \frac{(M-M_k)!(M_k-1)!}{(M_1-1)!\cdots(M_t-1)!} \tag{4.12}$$

$$= \frac{(M-M_k)!(M_k-1)!}{M!} \cdot \sum_{i=1}^{M} \sum_{D \in \mathscr{F}_i(M_k)} P(\mathscr{E}_i(D)|f(i)) \tag{4.13}$$

$$= \frac{(M-M_k)!(M_k-1)!}{M!} \cdot \sum_{D \in \mathscr{F}(M_k)} \sum_{i \in D} P(\mathscr{E}_i(D)|f(i)) \tag{4.14}$$

$$\geq \frac{(M-M_k)!(M_k-1)!}{M!} \cdot \sum_{D \in \mathscr{F}(M_k)} M_k P_e(K, M_k, N, Q) \tag{4.15}$$

$$= P_e(K, M_k, N, Q). \tag{4.16}$$

Corollary 4.1: For every DMC $K = (P, X, Y)$, every code f with fixed composition Q, block length N, and number of codewords M, and every positive integer H such that $M \geq 4H$,

$$\lambda(K, f) \geq (M/(2H)) P_e(K, H, N, Q). \tag{4.17}$$

Proof: Let M and H be such that $M \geq 4H$. If $H = 1$, then (4.17) is obviously true, because $P_e(K, H, N, Q) = 0$. So, without loss of generality, assume that $H \geq 2$. Let t and r be the unique integers such that $t \geq 1$, $0 \leq r < H-1$, and $M - 1 = t(H-1) + r$. Let $M_i = H$ for $i = 1,\cdots,t-1$ and $M_t = H + r$. The integers t, $M_1,\cdots,M_t$ satisfy the conditions of Lemma 4.1, so

$$\lambda(K, f) \geq (t-1) P_e(K, H, N, Q) + P_e(K, H+r, N, Q) \tag{4.18}$$

or

$$\lambda(K, f) \geq (t-1) P_e(K, H, N, Q). \tag{4.19}$$

By simple algebra, one can show that $M \geq 4H$ implies $(t-1) \geq M/(2H)$. Substitution of this inequality into (4.19) completes the proof.

We say that a code g is a *subcode* of a code f if the codewords of g form a subset of the codewords of f. A lower bound on $\lambda(K, f)$, which will be useful later, is the following.

Lemma 4.2: For every code f (not necessarily a fixed-composition code) and every subcode g of f,

$$\lambda(K, f) \geq (L/M) \lambda(K, g) \tag{4.20}$$

where L and M are the number of codewords in g and f, respectively.

Proof: Assume without loss of generality that the codewords of g are the first L codewords of f; in other words, $g(i) = f(i)$, $1 \leq i \leq L$. Then

$$\lambda(K, f) = \sum_{i=1}^{M} (1/M) \sum_{j=1}^{M} P(\mathscr{B}(i, j)|f(i)) \tag{4.21}$$

$$\geq \sum_{i=1}^{L} (1/M) \sum_{j=1}^{L} P(\mathscr{B}(i, j)|g(i)) \tag{4.22}$$

$$= (L/M) \lambda(K, g). \tag{4.23}$$

V. Sphere-Packing Lower Bounds for Fixed-Composition Codes

This section lists certain results[4] about lower bounds to the probability of decoding error for block codes. These results will be used in the next section to prove that $R_{\text{comp}} \leq R_0$. References to the proofs of results listed here can be found in the Appendix.

Let (P, X, Y) and (V, X, Y) be DMC's, and Q a p.d. on X. *Mutual information* $I(Q, V)$ and *informational divergence* $D(V\|P|Q)$ are functions defined as

$$I(Q, V) = \sum_{\xi \in X} \sum_{\eta \in Y} Q(\xi) V(\eta|\xi) \cdot \ln \left\{ V(\eta|\xi) \Big/ \sum_{\zeta \in X} Q(\zeta) V(\eta|\zeta) \right\} \tag{5.1}$$

$$D(V\|P|Q) = \sum_{\xi \in X} \sum_{\eta \in Y} Q(\xi) V(\eta|\xi) \ln \{ V(\eta|\xi)/P(\eta|\xi) \}. \tag{5.2}$$

For a DMC $K = (P, X, Y)$, a real number $R \geq 0$, and a p.d. Q on X, the *sphere-packing exponent* $E_{\text{sp}}(K, R, Q)$ is defined as

$$E_{\text{sp}}(K, R, Q) = \min_{V\colon R \geq I(Q,V)} D(V\|P|Q). \tag{5.3}$$

Lemma 5.1 (sphere-packing lower bound): For every $R > 0$, $\delta > 0$, every DMC $K = (P, X, Y)$, every pair of integers

[4] The results in this section are well-known, so we do not make an effort here to assign credit to original contributors. Our main reference is the book by Csiszar and Körner [8].

N and M, and every composition class Q on X^N,

$$P_e(K, M, N, Q) \geq (1/4)\exp\{-NE_{sp}(K, R, Q)(1+\delta)\} \tag{5.4}$$

whenever $(M-1)/2 \geq \exp N(R+\delta)$ and $N \geq N_0(|X|, |Y|, \delta)$. The function N_0 is independent of Q and finite for all $\delta > 0$.

Lemma 5.2 (some properties of $E_{sp}(K, R, Q)$): For every fixed DMC $K = (P, X, Y)$ and p.d. Q on X, $E_{sp}(K, R, Q)$ is a convex nonincreasing function of $R \geq 0$. $E_{sp}(K, R, Q)$ is positive for $0 \leq R < I(Q, P)$ and zero for $R \geq I(Q, P)$. There is a rate $R_c(K, Q)$, called the *critical rate* for Q, which has the property that

$$R_c(K, Q) + E_{sp}(K, R_c(K, Q), Q) = E_0(K, Q) \tag{5.5}$$

where[5]

$$E_0(K, Q) = \min_V D(V\|P|Q) + I(Q, V). \tag{5.6}$$

Lemma 5.3:

$$\max_Q E_0(K, Q) = R_0(K), \qquad \text{for all } K. \tag{5.7}$$

Lemma 5.4:

$$\max_Q R_c(K, Q) \leq R_0(K), \qquad \text{for all } K. \tag{5.8}$$

In (5.7) and (5.8), the maximum is taken over all p.d.'s on the input alphabet of K, and $R_0(K)$ is as defined in (1.1).

VI. Proof that R_0 Upper-Bounds R_{comp}

$R_{comp} \leq R_0$ follows immediately from the following lemma.

Lemma 6.1: Let $\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots$ be a sequence of block codes for a DMC $K = (P, X, Y)$. Let N_i denote the block length of $\boldsymbol{f}_i$ and M_i the number of codewords in $\boldsymbol{f}_i$. Assume that N_i increases monotonically with i and that there exists some $\epsilon > 0$ such that $M_i \geq \exp\{N_i(R_0(K) + \epsilon)\}$ for all i. Then, for all i sufficiently large,

$$\lambda(K, \boldsymbol{f}_i) \geq \exp\{N_i\epsilon/8\}. \tag{6.1}$$

Proof: (To simplify the notation, we suppress the dependence of functions on K in this proof.) Since $(1+N_i)^{|X|}$ is an upper bound on the number of composition classes on X^{N_i} [8, p. 29], there exists a fixed-composition subcode of $\boldsymbol{f}_i$ with at least $M_i/(1+N_i)^{|X|}$ codewords. Let $\boldsymbol{g}_i$ be such a subcode of $\boldsymbol{f}_i$, L_i the number of codewords in $\boldsymbol{g}_i$, and Q_i the composition of $\boldsymbol{g}_i$. Let us define

$$\delta = \epsilon/(8 + 4R_0). \tag{6.2}$$

Claim: There is a function $\Omega(\epsilon)$ such that for all $i \geq \Omega(\epsilon)$ the following conditions hold simultaneously.

1. $N_i \geq N_0(|X|, |Y|, \delta)$. (The function N_0 here is the same as the N_0 in Lemma 5.1.) (6.3)
2. a) $(1/N_i)\ln L_i > R_0 + \epsilon/2$ (6.4)
   b) $(1/N_i)\ln(L_i/8M_i) \geq -\epsilon/8$. (6.5)
3. There exist integers H_i such that
   a) $L_i \geq 4H_i$ (6.6)
   b) $R_c(Q_i) + \delta < (1/N_i)\ln((H_i - 1)/2)$ (6.7)
   c) $R_c(Q_i) + 2\delta > (1/N_i)\ln H_i$. (6.8)

Proof of the Claim: First, it is obvious that (6.3) is satisfied for all i sufficiently large, because $N_0(|X|, |Y|, \delta)$ is finite and independent of Q_i.

For (6.4) and (6.5), simply verify that $(1/N_i)\ln(M_i/L_i)$ tends to zero as i goes to infinity, and recall that $M_i \geq \exp\{N_i(R_0 + \epsilon)\}$.

For (6.6)–(6.8), first verify that the difference of the right sides of (6.7) and (6.8) tends to zero as i goes to infinity, and conclude that conditions (6.7) and (6.8) essentially amount to having

$$R_c(Q_i) + \delta < (1/N_i)\ln H_i < R_c(Q_i) + 2\delta \tag{6.9}$$

for all sufficiently large i. Now, clearly, a number H_i, satisfying (6.9) and (6.6), can be found since $R_c(Q_i) + 2\delta < (1/N_i)\ln L_i$ for all sufficiently large i. This is because, first, by the argument in the preceding paragraph, $R_0 + \epsilon/2 < (1/N_i)\ln L_i$ for all sufficiently large i, and second, by Lemma 5.4 and (6.2), $R_c(Q_i) + 2\delta \leq R_0 + \epsilon/4$ for all i.

Now suppose that $i \geq \Omega(\epsilon)$. Let H_i be chosen so that (6.6)–(6.8) are satisfied. The proof is completed by the following sequence of steps, each of which is subsequently justified:

$$\lambda(K, \boldsymbol{f}_i)$$

$$\geq (L_i/M_i)\lambda(K, \boldsymbol{g}_i) \tag{6.10}$$

$$\geq \left(L_i^2/(2M_iH_i)\right)P_e(K, H_i, N_i, Q_i) \tag{6.11}$$

$$\geq \left(L_i^2/(8M_iH_i)\right)\exp\{-N_iE_{sp}(R_c(Q_i), Q_i)(1+\delta)\} \tag{6.12}$$

$$= \left(L_i^2/(8M_iH_i)\right)\exp\{-N_i(1+\delta)[E_0(Q_i) - R_c(Q_i)]\} \tag{6.13}$$

$$\geq \left(L_i^2/(8M_iH_i)\right)\exp\{-N_i(1+\delta)[R_0 - R_c(Q_i)]\} \tag{6.14}$$

$$\geq (L_i/8M_i)\exp\{-N_i\{(1+\delta)[R_0 - R_c(Q_i)] - R_0 - \epsilon/2 + R_c(Q_i) + 2\delta\}\} \tag{6.15}$$

$$= (L_i/8M_i)\exp\{N_i\{\epsilon/2 - 2\delta - \delta R_0 + \delta R_c(Q_i)\}\} \tag{6.16}$$

$$\geq (L_i/8M_i)\exp\{N_i(\epsilon/2 - 2\delta - \delta R_0)\} \tag{6.17}$$

$$= (L_i/8M_i)\exp\{N_i\epsilon/4\} \tag{6.18}$$

$$\geq \exp\{N_i\epsilon/8\}. \tag{6.19}$$

Inequality (6.10) follows by Lemma 4.2; (6.11) by (6.6) and Corollary 4.1; (6.12) by (6.3), (6.7), and Lemma 5.1; (6.13) by Lemma 5.2; (6.14) by Lemma 5.3; (6.15) by (6.4) and (6.8); (6.17) by the nonnegativity of $R_c(Q_i)$; (6.18) by (6.2); and (6.19) by (6.5).

Theorem: For every discrete memoryless channel K,

$$R_0(K) \geq R_{comp}(K). \tag{6.20}$$

[5] The function E_0 here and the E_0 in Section I are different functions.

Proof: If $\boldsymbol{e}$ is a tree code for K with rate $> R_0(K)$, then by Lemma 6.1, $\lambda(K, \boldsymbol{e}(t))$ increases *exponentially* in t; hence by Lemma 3.1, so do $\Lambda(K, \boldsymbol{e}, \Gamma, t)$ and $(1/t)\Lambda(K, \boldsymbol{e}, \Gamma, t)$. Therefore, rates above $R_0(K)$ are not achievable by sequential decoding; i.e., $R_0(K) \geq R_{\text{comp}}(K)$.

VII. Complementary Remarks

To state a fundamental property of R_0, let us recall a result, known as the union or the Bhattacharyya bound (see, e.g., [12, p. 68]), which constitutes a converse to Lemma 6.1. For every pair of positive integers M and N, there exists a block code $\boldsymbol{f}$ with block length N and number of codewords M such that

$$\lambda(K, \boldsymbol{f}) \leq M \exp\{-NR_0(K)\}. \tag{7.1}$$

In view of Lemma 6.1 and (7.1), $R_0(K)$ can be interpreted as a threshold such that for every $R \geq 0$: 1) if $R > R_0$, then, for every sequence of block codes $\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3, \cdots$ with increasing block length, and rate $\geq R$ for each $\boldsymbol{f}_k$, $\lambda(K, \boldsymbol{f}_k)$ goes to infinity (exponentially) with increasing k; 2) if $R < R_0$, then there exists a sequence of block codes $\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3, \cdots$ with increasing block length, and rate $\geq R$ for each $\boldsymbol{f}_k$, such that $\lambda(K, \boldsymbol{f}_k)$ goes to zero (exponentially) with increasing k. (The behavior of λ is in general unknown for $R = R_0$.)

It is important to note that the above property of R_0 makes no mention of tree codes or sequential decoding, while the cutoff rate R_{comp} is defined in terms of tree codes and sequential decoding. Strictly speaking, it is inappropriate to refer to R_0 as the cutoff rate, even though it turns out that $R_0 = R_{\text{comp}}$; instead, R_0 should be recognized as a threshold in the above sense.

R_0 upper-bounds R_{comp} essentially because of property 1) above, and the fact that sequential decoding does not have a "look-ahead" capability. It would be incorrect to think that R_0, because of the above properties, constitutes a limit for *every* system to rates at which reliable communication is possible in *practice*. There are many well-known practical decoding schemes that can achieve rates well beyond R_0. Some of these schemes are in fact variations of sequential decoding itself. As a recent example, we may cite the work of Massey [9] on a certain optical channel.

It is of theoretical interest to determine the exact exponential rate of increase of the quantities $\lambda(K, \boldsymbol{f}_i)$ in Lemma 6.1, that is, the behavior of $(1/N_i)\ln \lambda(K, \boldsymbol{f}_i)$ as i goes to infinity. With minor modifications to the proof of Lemma 6.1, one can prove that under the hypotheses of Lemma 6.1 and for any $\epsilon' > 0$,

$$\lambda(K, \boldsymbol{f}_i) \geq \exp\{N_i \epsilon(1-\epsilon')\} \tag{7.2}$$

for all sufficiently large i (i.e., $i \geq \Omega(\epsilon, \epsilon')$ for some function Ω). This can be interpreted as saying that the exponential rate of increase of λ is never smaller than the excess rate over R_0.

Another problem of interest is to determine whether instantaneous feedback from channel output to encoder input increases the cutoff rate (or the R_0 threshold). Preliminary work shows that feedback does not increase the cutoff rate, at least for a class of symmetric DMC's which includes the binary symmetric channel. This subject will be explored further in a future publication.

Acknowledgment

I would like to express my gratitude to Professor Robert G. Gallager for invaluable guidance and support throughout this work.

Appendix

This appendix provides references to proofs of results listed in Section V.

The *maximum* probability of decoding error for a DMC K, a block code $\boldsymbol{f}$, and a decoder $\boldsymbol{d}$ is defined as

$$P_{e,\max}(K, \boldsymbol{f}, \boldsymbol{d}) = \max_i \Pr\{z \neq i | \boldsymbol{f}(i)\} \tag{A.1}$$

where z denotes the decoder output and the maximum is taken over all messages (cf. (4.1)).

Lemma 5.1 follows from the following theorem.

Theorem [8, p. 166]: For every $R > 0$, $\delta > 0$, every DMC $K = (P, X, Y)$, every fixed-composition code $\boldsymbol{f}$ with block length N, number of codewords M, and composition Q, and every decoder $\boldsymbol{d}$, the maximum probability of error satisfies

$$P_{e,\max}(K, \boldsymbol{f}, \boldsymbol{d}) \geq (1/2)\exp\{-NE_{\text{sp}}(K, R, Q)(1+\delta)\} \tag{A.2}$$

whenever $M \geq \exp N(R+\delta)$ and $N \geq n_0(Q, |X|, |Y|, \delta)$.

Here, the function n_0 is finite for all Q and all $\delta > 0$. The function N_0 in Lemma 5.1 can be taken as

$$N_0(|X|, |Y|, \delta) = \sup_Q n_0(Q, |X|, |Y|, \delta) \tag{A.3}$$

where the supremum is over all composition classes on X. The reader is invited to examine the origin of the function n_0 in [8] to verify that N_0, as defined above, is finite for all $\delta > 0$.

The argument involved in going from the above theorem (a lower bound on the maximum probability of error) to Lemma 5.1 (a lower bound on the average probability of error) is well-known [10, eq. 4.41] and will be omitted.

The assertions of Lemma 5.2 are contained in [8, lemma 5.4 and corollary 5.4, p. 168].

Lemma 5.3 is from [8, problem 23, p. 192]. Its proof, based on hints given in [8], can be found in [11].

Proof of Lemma 5.4: $R_c(K, Q) \leq E_0(K, Q)$ by Lemma 5.2, and $E_0(K, Q) \leq R_0(K)$ by Lemma 5.3. Hence, $R_c(K, Q) \leq R_0(K)$ for all K and Q.

References

[1] J. M. Wozencraft, "Sequential Decoding for Reliable Communications," Tech. Rep. 325, RLE, Massachusetts Institute of Technology, Cambridge, MA, 1957.

[2] R. M. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inform. Theory*, vol. IT-9, pp. 64–74, Apr. 1963.

[3] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.

[4] I. M. Jacobs and E. R. Berlekamp, "A lowerbound to the distribution of computation for sequential decoding," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 167–174, Apr. 1967.
[5] J. E. Savage, "Sequential decoding—the computation problem," *Bell Syst. Tech. J.*, vol. 45, no. 1, pp. 149–175, 1966.
[6] K. Zigangirov, "Some sequential decoding procedures," *Problemy Peredachi Inf.*, vol. 2, pp. 13–25, 1966.
[7] F. Jelinek, "A fast sequential decoding algorithm using a stack," *IBM J. Res. Dev.*, vol. 13, pp. 675–685, 1969.
[8] I. Csiszar and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Channels*. New York: Academic, 1981.
[9] J. L. Massey, "Capacity, cutoff rate, and coding for a direct-detection optical channel," *IEEE Trans. Commun.*, vol. COM-29, pp. 1615–1621, Nov. 1981.
[10] C. E. Shannon, R. G. Gallager, and E. R. Berlekamp, "Lowerbounds to error probability for coding on discrete memoryless channels," Parts I and II, *Information and Control*, vol. 10, pp. 65–103 and pp. 522–552, 1967.
[11] E. Arikan, "Sequential Decoding for Multiple Access Channels," Ph.D. dissertation, MIT, Cambridge, MA, Nov. 1985.
[12] R. J. McEliece, *The Theory of Information and Coding*. Reading, MA: Addison-Wesley, 1977.